The SciDB Approach

and its applicability to HEP computing

Kian-Tat Lim

SLAC National Accelerator Laboratory

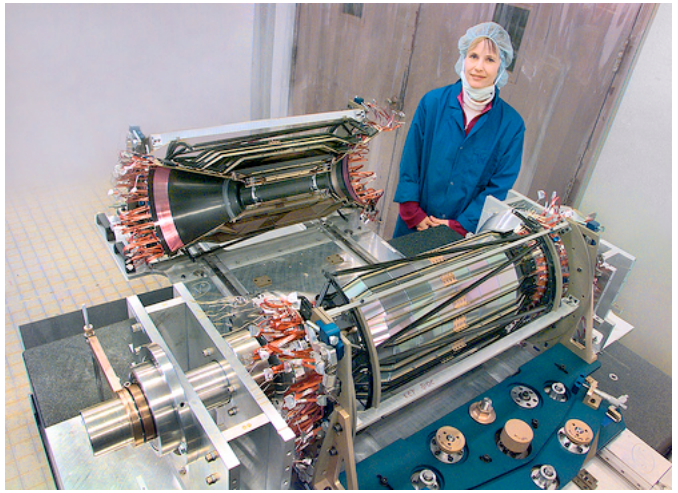
SLAC

Who Am I Not?

Image Credits:
Flickr User thetoad01,
Lawrence Berkeley Lab

Salesperson

High-energy physicist



Who Am I?

Image Credit:
Flickr User cdixon

Long experience with large data

Both science and industry perspective

SLAC NATIONAL ACCELERATOR LABORATORY

LSST

YAHOO!

Outline

Why use a database?

SciDB

SciDB applicability

Lessons learned

Why use a database?

Storage?

Query

Analysis

Query

Extract small subsets

Analysis

Process large sets:

Aggregates
Large subsets
Pairwise analysis
Data mining

Efficiency

High-level language

Schema

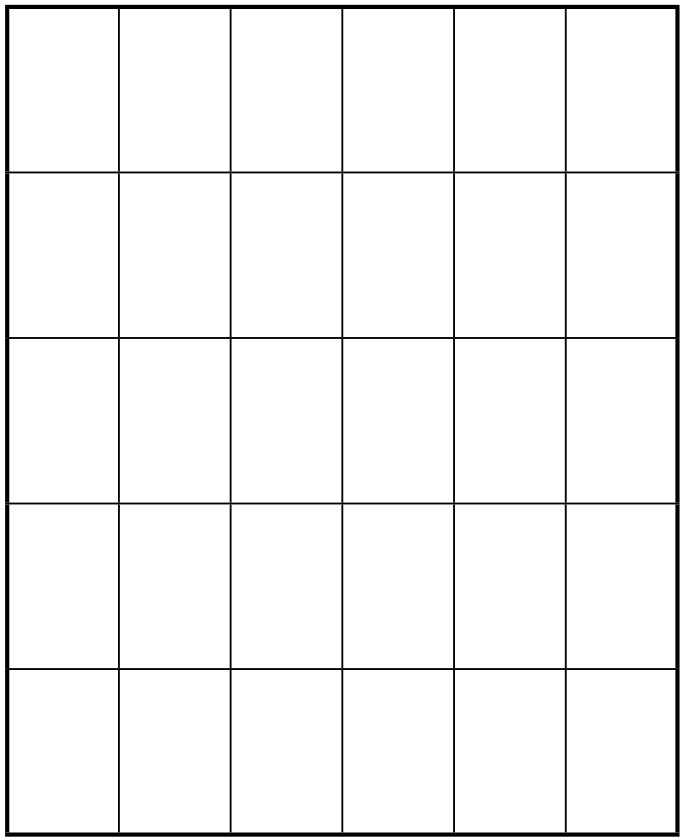
Transactions

Parallelism

Data Model

Relations = sets of tuples

Must fit well



Outline

Why use a database?

SciDB

SciDB applicability

Lessons learned

SciDB

History

Data model

Features

SciDB History

XLDB1

Science/Academia Meeting

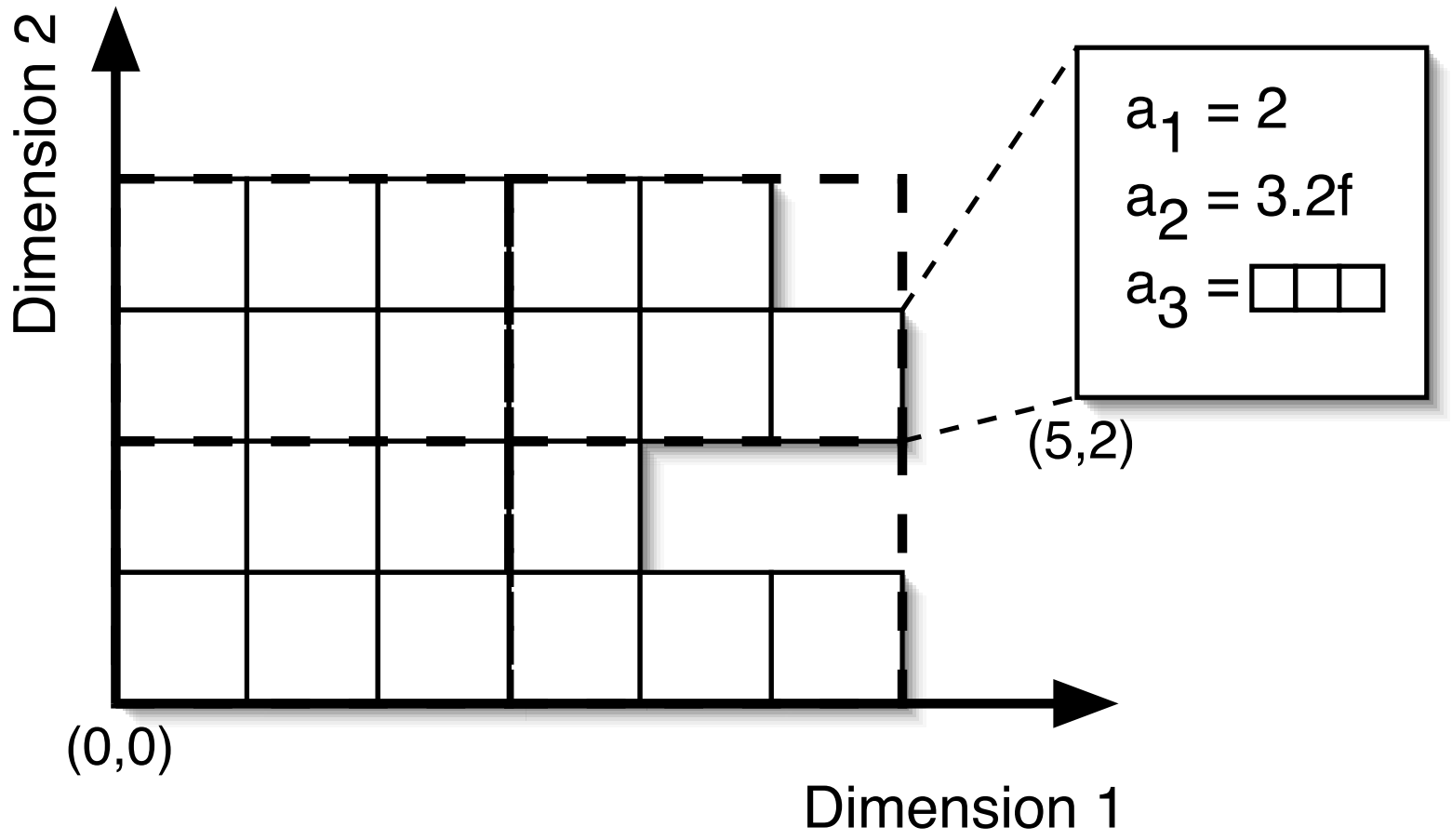
XLDB2

VLDB + XLDB3

SciDB Data Model

Array-based model,
attribute-based storage

Array Data Model



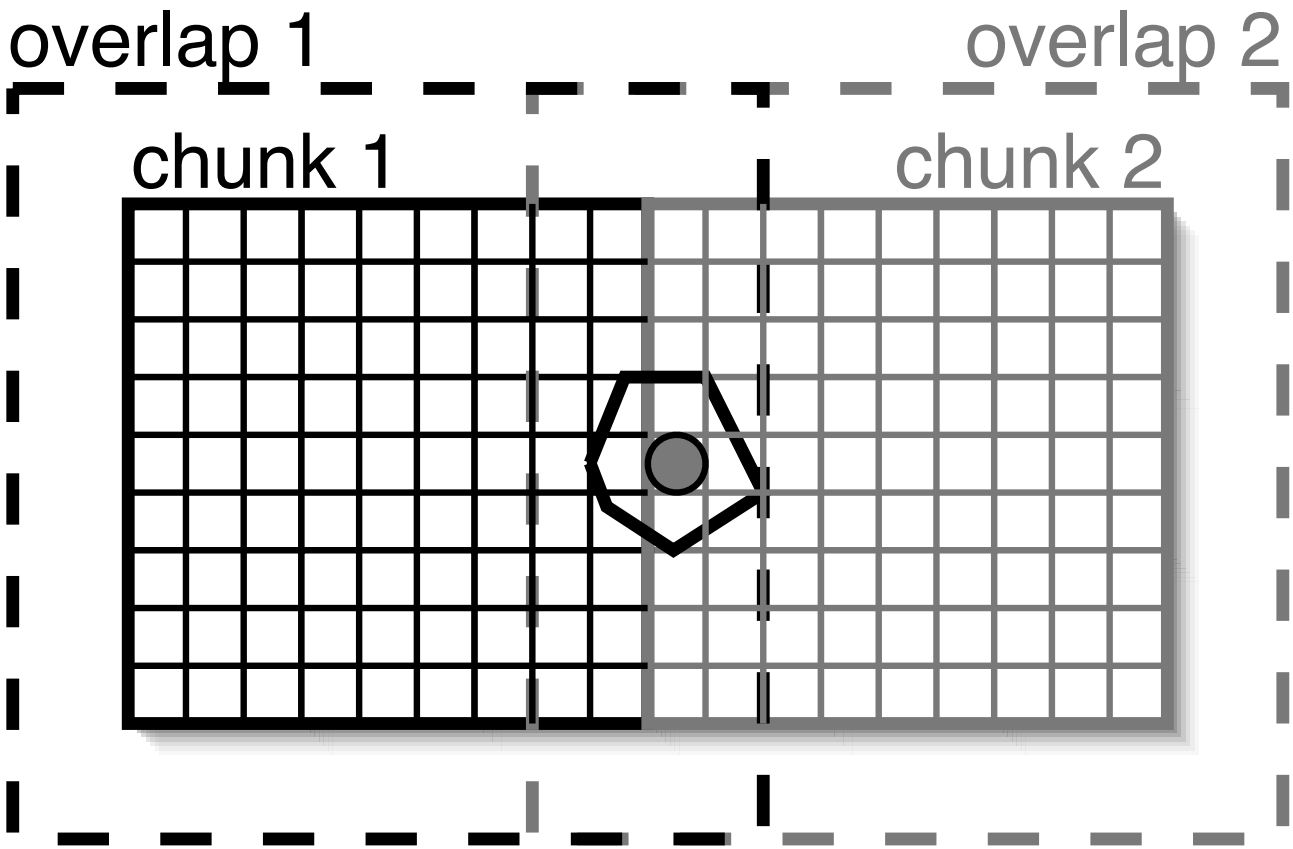
Why Arrays?

Familiar interface

Ordering

Physical relationships

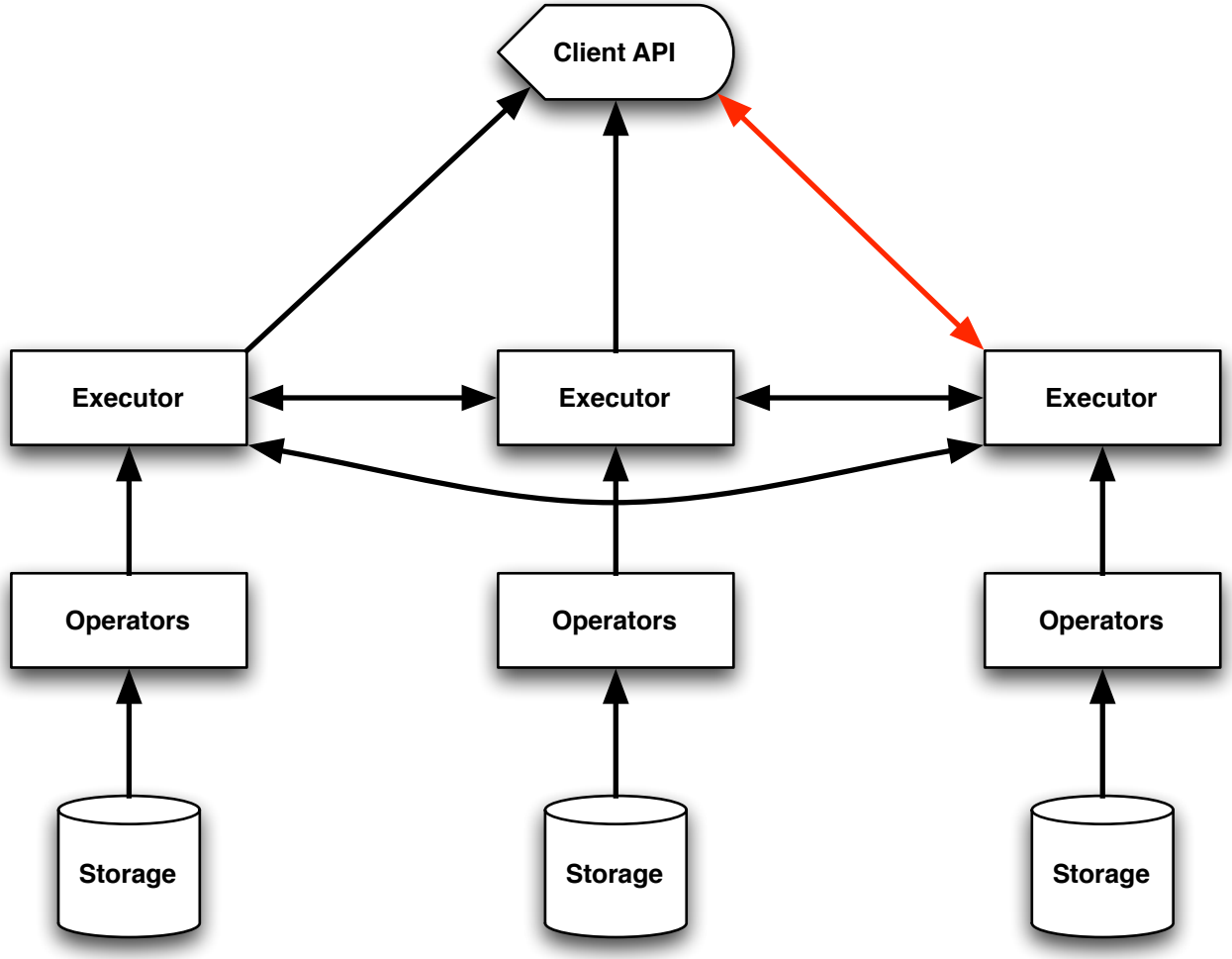
Overlapping Chunks



SciDB Features

Parallel, scalable, elastic, extensible

UDFs, UDTs, versioning, provenance, uncertainty, ...


Client API
Executor
Executor
Executor
Operators
Operators
Operators
Storage
Storage
Storage

Outline

Why use a database?

SciDB

SciDB applicability

Lessons learned

SciDB and HEP Events

Arrays vs. sets?

Tuples vs. trees?

Joins vs. pointers?

SciDB and HEP Conditions

More regularly structured

Mostly retrieval, not analytic?

Outline

Why use a database?

SciDB

SciDB applicability

Lessons learned

Lessons Learned

Fault tolerance

Computation location

Data loading/storage

Bandwidth vs. CPU

Per-user data

Fault Tolerance

Failures are routine

Build in support early

Software beats hardware

Computation Location

Moving a PB is expensive
Move compute to data

Data Loading/Storage

Loading is movement

Replicas are cheap

Dynamic translators

Bandwidth vs. CPU

Many spindles

Retrieve only necessary data

Compression

Sequential I/O vs. seeks

Per-User Data

Data annotation
Derived products
Uploads

Conclusion

SciDB model

SciDB lessons